

dnS2exy

a verifying DNS (SEc) proXY

Willem Toorop

Willem@NLnetLabs.nl



18 April 2012

Motivation

Prevent bogus zones getting out in the wild.

Motivation

Prevent bogus zones getting out in the wild.

- There are quite a few DNS(SEC) verifiers available

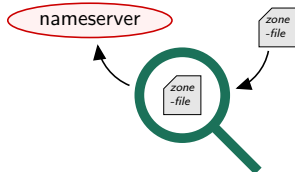
OpenDNSSEC Auditor	.SE dnssec-monitor	jdnssec-tools
Keychecker	nagval	SurfNet DNSSEC Checker
validns	Vantages D-Sync	ldns-verify-zone
YAZVS	AFNIC ZoneCheck	*

*list taken from the MEN&MICE website

Motivation

Prevent bogus zones getting out in the wild.

- ▶ There are quite a few DNS(SEC) verifiers available
- ▶ But they need to check the *zone-file* before it is served



OpenDNSSEC Auditor .SE dnssec-monitor jdnssec-tools
Keychecker nagval SurfNet DNSSEC Checker
validns Vantages D-Sync Idns-verify-zone
YAZVS AFNIC ZoneCheck

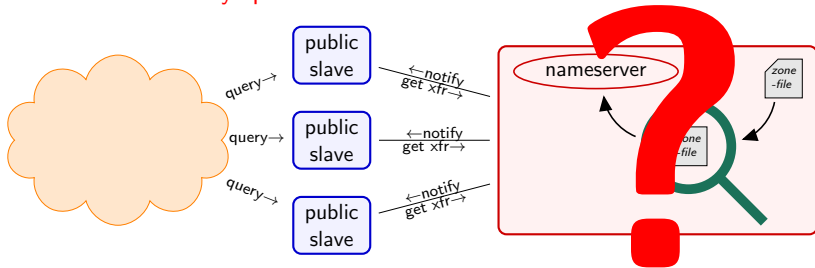
*

*list taken from the MEN&MICE website

Motivation

Prevent bogus zones getting out in the wild.

- ▶ There are quite a few DNS(SEC) verifiers available
- ▶ But they need to check the *zone-file* before it is served
 - ▶ Not always possible ...



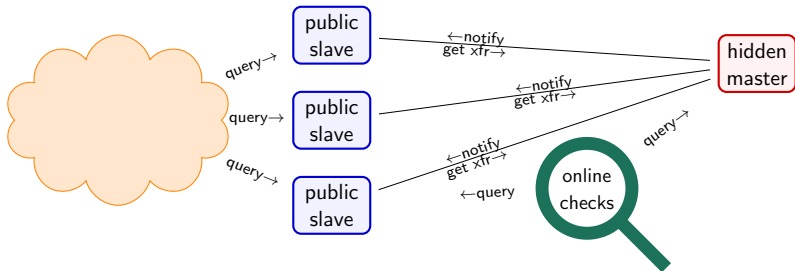
OpenDNSSEC Auditor .SE dnssec-monitor jdnssec-tools
Keychecker nagval SurfNet DNSSEC Checker
validns Vantages D-Sync Idns-verify-zone
YAZVS AFNIC ZoneCheck

*list taken from the MEN&MICE website

Motivation

Prevent bogus zones getting out in the wild.

- ▶ There are quite a few DNS(SEC) verifiers available
- ▶ But they need to check the *zone-file* before it is served
- ▶ Or query a zone and thus detect flaws post-factum

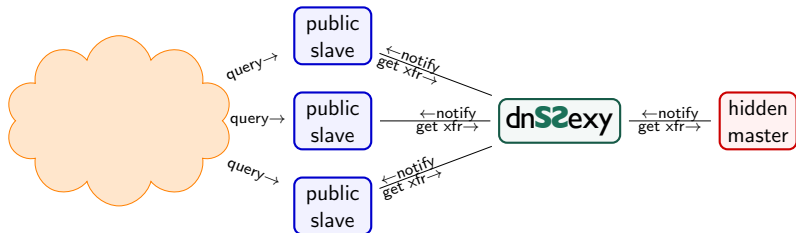


OpenDNSSEC Auditor .SE dnssec-monitor jdnssec-tools
Keychecker nagval SurfNet DNSSEC Checker
validns Vantages D-Sync Idns-verify-zone
YAZVS AFNIC ZoneCheck

*list taken from the MEN&MICE website

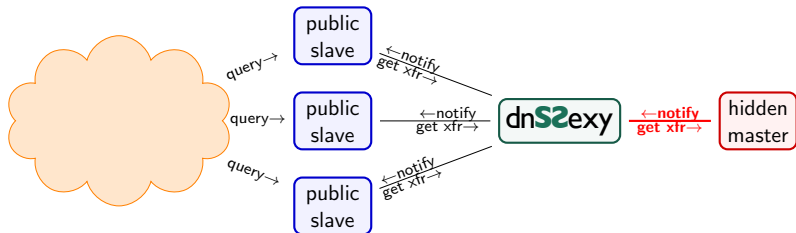
dnS2exy a DNS (SEc) proXY

- ▶ dnS2exy is just another nameserver that sits between the hidden master and the public slaves



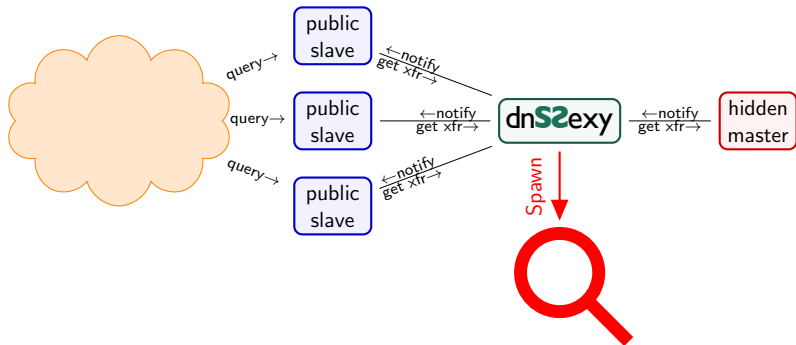
dnS2exy a DNS (SEc) proXY

- ▶ dnS2exy is just another nameserver that sits between the hidden master and the public slaves
- ▶ On update dnS2exy is notified, transfers, but does not serve yet



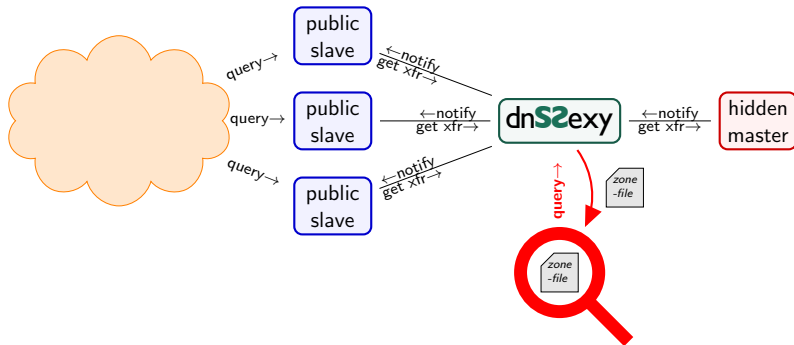
dnS2exy a DNS (SEc) proXY

- ▶ dnS2exy is just another nameserver that sits between the hidden master and the public slaves
- ▶ On update dnS2exy is notified, transfers, but does not serve yet
- ▶ First a process is spawn for verifying the zone



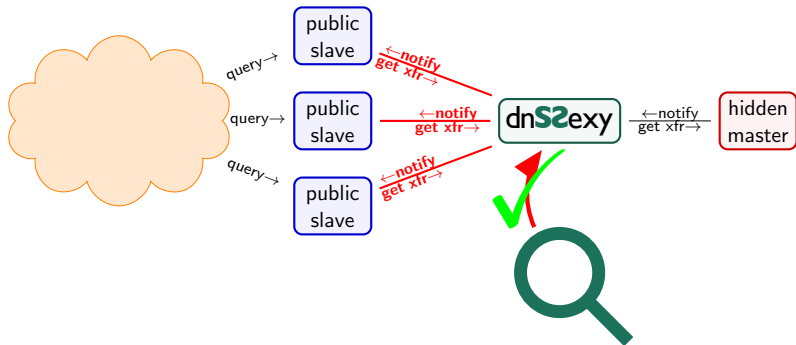
dnS2exy a DNS (SEc) proXY

- ▶ dnS2exy is just another nameserver that sits between the hidden master and the public slaves
- ▶ On update dnS2exy is notified, transfers, but does not serve yet
- ▶ First a process is spawn for verifying the zone
- ▶ The updated zone is fed and served to the verifier



dnS2exy a DNS (SEc) proXY

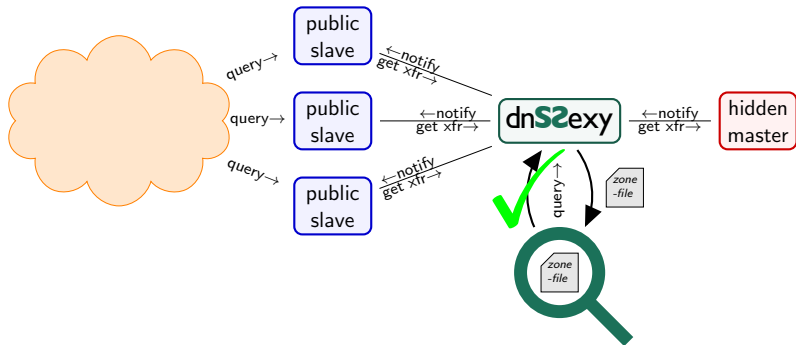
- ▶ dnS2exy is just another nameserver that sits between the hidden master and the public slaves
- ▶ On update dnS2exy is notified, transfers, but does not serve yet
- ▶ First a process is spawn for verifying the zone
- ▶ The updated zone is fed and served to the verifier
- ▶ Only when the verifying process exits succesfully, the zone is loaded and served.



dnS2exy a DNS (SEc) proXY

OpenDNSSEC Auditor .SE dnssec-monitor jdnssec-tools
Keychecker nagval SurfNet DNSSEC Checker
validns Vantages D-Sync Idns-verify-zone
YAZVS AFNIC ZoneCheck

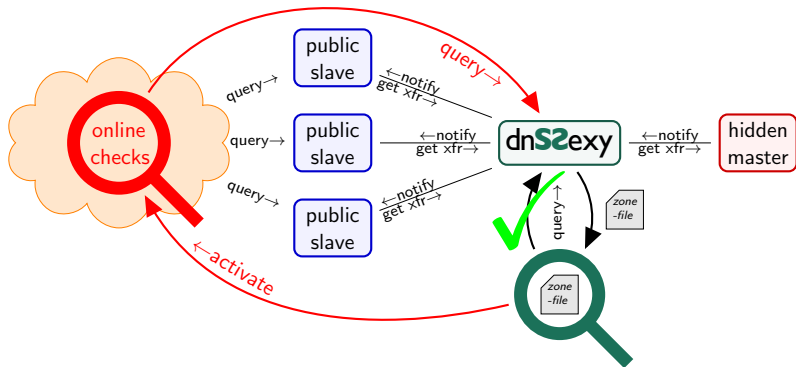
- Can use both **querying** and **zonefile** reading verifiers **pre-factum**



dnS2exy a DNS (SEc) proXY

OpenDNSSEC Auditor .SE dnssec-monitor jdnssec-tools
Keychecker nagval SurfNet DNSSEC Checker
validns Vantages D-Sync Idns-verify-zone
YAZVS AFNIC ZoneCheck

- ▶ Can use both **querying** and **zonefile** reading verifiers **pre-factum**
- ▶ Or even use online verifying services



Implementation

dnS2exy is a fork of NSD version 3.2

- ▶ `svn co https://open.nlnetlabs.nl/svn/nsd/branches/NSD_3_2_SEXY`
- + Robust proven nameserver
- + All facilities for notify/xfr already there
- + Fork-kill modus operandi suits dnS2exy well!

Implementation

dnS2exy is a fork of NSD version 3.2

- ▶ `svn co https://open.nlnetlabs.nl/svn/nsd/branches/NSD_3_2_SEXY`
- + Robust proven nameserver
- + All facilities for notify/xfr already there
- + Fork-kill modus operandi suits dnS2exy well!

Zone Options

For every zone the options need to be specified in one zone: clause. The access control list elements can be given multiple times to add multiple servers. These elements need to be added explicitly.

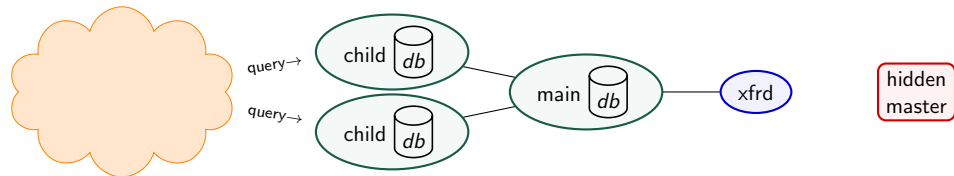
```
name: <string>
zonefile: <filename>

allow-notify: <ip-spec> <key-name | NOKEY | BLOCKED>
request-xfr: [AXFR|UDP] <ip-address> <key-name | NOKEY>
notify: <ip-address> <key-name | NOKEY>
provide-xfr: <ip-spec> <key-name | NOKEY | BLOCKED>
```

Implementation

dnS2exy is a fork of NSD version 3.2

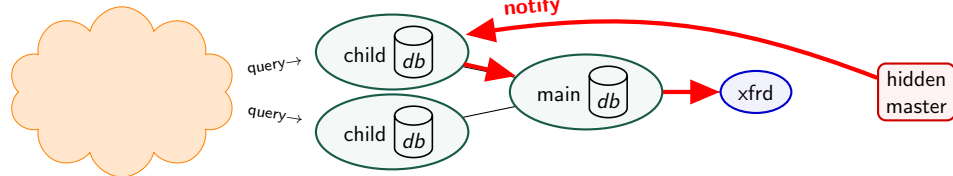
- ▶ `svn co https://open.nlnetlabs.nl/svn/nsd/branches/NSD_3_2_SEXY`
- + Robust proven nameserver
- + All facilities for notify/xfr already there
- + Fork-kill modus operandi suits dnS2exy well!



Implementation

dnS2exy is a fork of NSD version 3.2

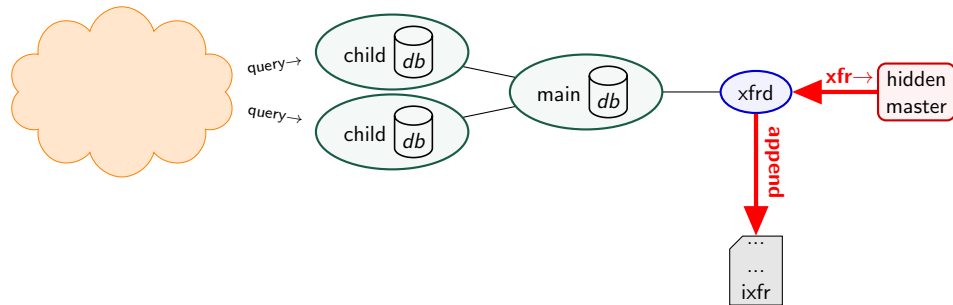
- ▶ `svn co https://open.nlnetlabs.nl/svn/nsd/branches/NSD_3_2_SEXY`
- + Robust proven nameserver
- + All facilities for notify/xfr already there
- + Fork-kill modus operandi suits dnS2exy well!



Implementation

dnS2exy is a fork of NSD version 3.2

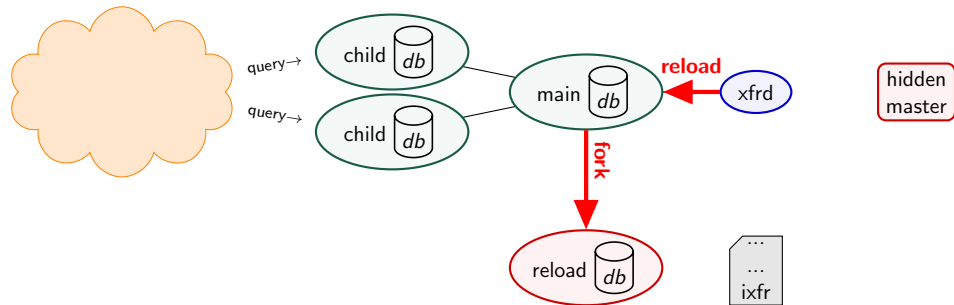
- ▶ `svn co https://open.nlnetlabs.nl/svn/nsd/branches/NSD_3_2_SEXY`
- + Robust proven nameserver
- + All facilities for notify/xfr already there
- + Fork-kill modus operandi suits dnS2exy well!



Implementation

dnS2exy is a fork of NSD version 3.2

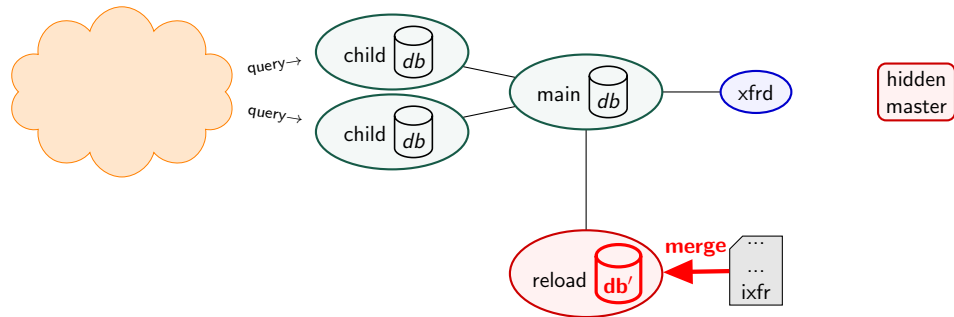
- ▶ `svn co https://open.nlnetlabs.nl/svn/nsd/branches/NSD_3_2_SEXY`
- + Robust proven nameserver
- + All facilities for notify/xfr already there
- + Fork-kill modus operandi suits dnS2exy well!



Implementation

dnS2exy is a fork of NSD version 3.2

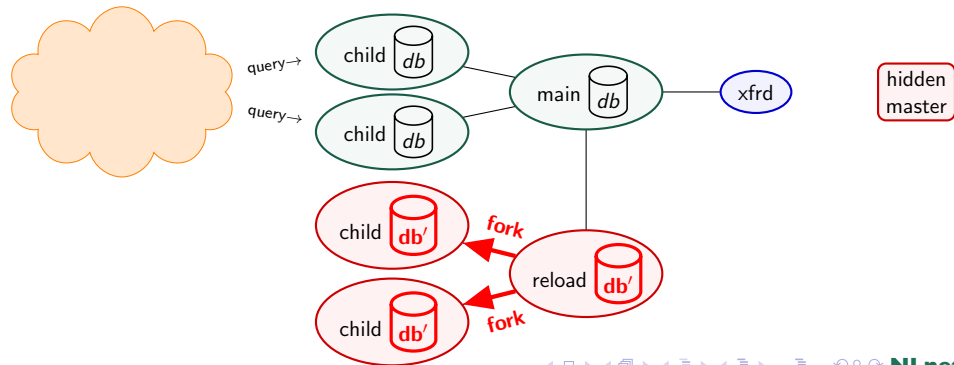
- ▶ `svn co https://open.nlnetlabs.nl/svn/nsd/branches/NSD_3_2_SEXY`
- + Robust proven nameserver
- + All facilities for notify/xfr already there
- + Fork-kill modus operandi suits dnS2exy well!



Implementation

dnS2exy is a fork of NSD version 3.2

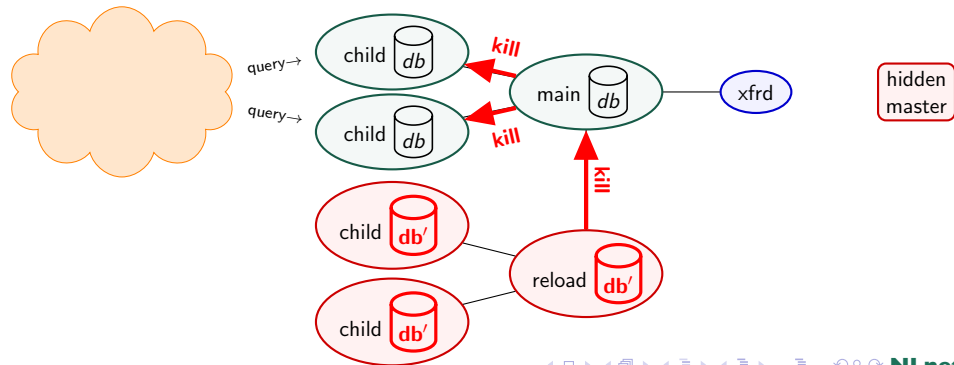
- ▶ `svn co https://open.nlnetlabs.nl/svn/nsd/branches/NSD_3_2_SEXY`
- + Robust proven nameserver
- + All facilities for notify/xfr already there
- + Fork-kill modus operandi suits dnS2exy well!



Implementation

dnS2exy is a fork of NSD version 3.2

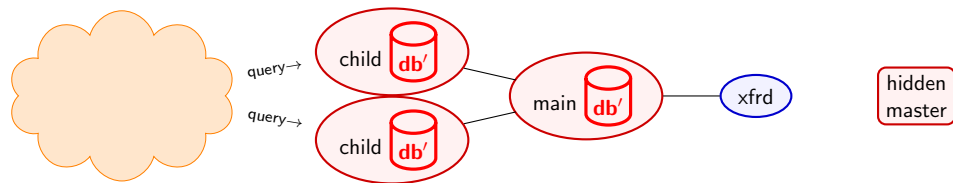
- ▶ `svn co https://open.nlnetlabs.nl/svn/nsd/branches/NSD_3_2_SEXY`
- + Robust proven nameserver
- + All facilities for notify/xfr already there
- + Fork-kill modus operandi suits dnS2exy well!



Implementation

dnS2exy is a fork of NSD version 3.2

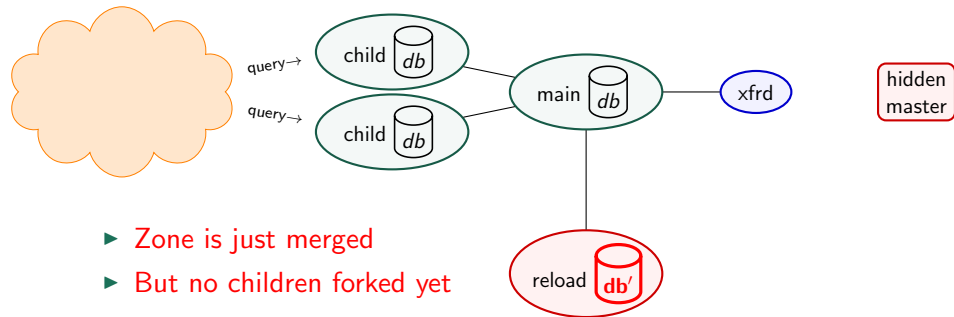
- ▶ `svn co https://open.nlnetlabs.nl/svn/nsd/branches/NSD_3_2_SEXY`
- + Robust proven nameserver
- + All facilities for notify/xfr already there
- + Fork-kill modus operandi suits dnS2exy well!



Implementation

dnS2exy is a fork of NSD version 3.2

- ▶ `svn co https://open.nlnetlabs.nl/svn/nsd/branches/NSD_3_2_SEXY`
- + Robust proven nameserver
- + All facilities for notify/xfr already there
- + Fork-kill modus operandi suits dnS2exy well!

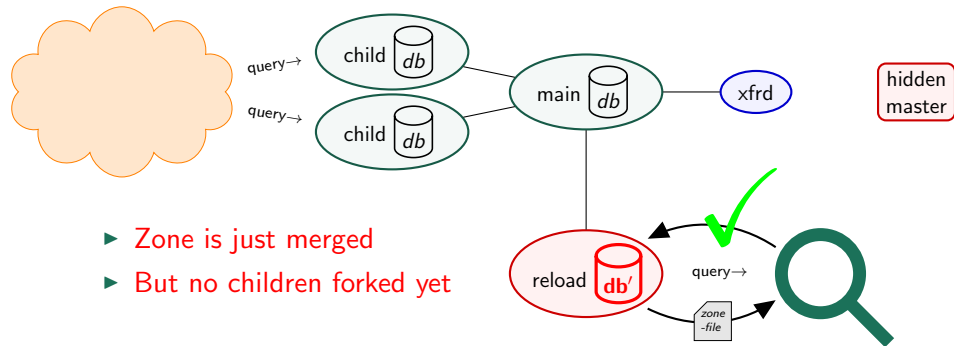


- ▶ Zone is just merged
- ▶ But no children forked yet

Implementation

dnS2exy is a fork of NSD version 3.2

- ▶ `svn co https://open.nlnetlabs.nl/svn/nsd/branches/NSD_3_2_SEXY`
- + Robust proven nameserver
- + All facilities for notify/xfr already there
- + Fork-kill modus operandi suits dnS2exy well!

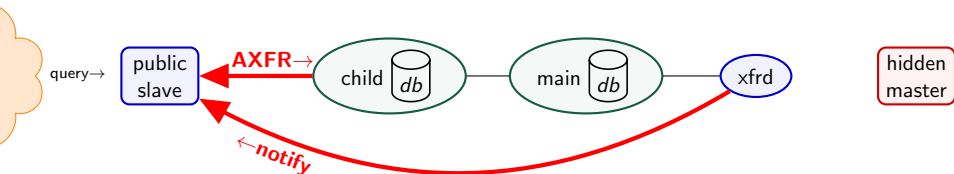


- ▶ Zone is just merged
- ▶ But no children forked yet

Implementation

dnS2exy is a fork of NSD version 3.2

- ▶ `svn co https://open.nlnetlabs.nl/svn/nsd/branches/NSD_3_2_SEXY`
- + Robust proven nameserver
- + All facilities for notify/xfr already there
- + Fork-kill modus operandi suits dnS2exy well!
- No incremental transfers out



Configuring dnS2exy

server:

zone:

name: <domain name>

allow-notify: <ip-spec> <key-name | NOKEY | BLOCKED>

request-xfr: [AXFR|UDP] <ip-address> <key-name | NOKEY>

verifier: <program with arguments>

► For example: `ldns-verify-zone`

VERIFY_ZONE

name of the changed zone

Configuring dnS2exy

server:

verifier-count: <# concurrently running verifiers>

zone:

name: <domain name>

allow-notify: <ip-spec> <key-name | NOKEY | BLOCKED>

request-xfr: [AXFR|UDP] <ip-address> <key-name | NOKEY>

verifier: <program with arguments>

VERIFY_ZONE

name of the changed zone

Configuring dnS2exy

server:

verifier-count: <# concurrently running verifiers>

verifier-timeout: <# seconds>

zone:

name: <domain name>

allow-notify: <ip-spec> <key-name | NOKEY | BLOCKED>

request-xfr: [AXFR|UDP] <ip-address> <key-name | NOKEY>

verifier: <program with arguments>

verifier-timeout: <# seconds | inherit>

VERIFY_ZONE

name of the changed zone

Configuring dnS2exy

server:

verifier-count: <# concurrently running verifiers>

verifier-timeout: <# seconds>

verify-ip-address: <ip4 or ip6>[@port]

- ▶ Needs to be set for verifiers that need to query
- ▶ Can be given multiple times

verify-port: <number>

- ▶ defaults to 5347, because: 5E87

zone:

verifier: <program with arguments>

verifier-timeout: <# seconds | inherit>

VERIFY_ZONE

name of the changed zone

VERIFY_IP_ADDRESSES

list of <address@port> values

VERIFY_IP_ADDRESS

first ip address in the list

VERIFY_PORT

VERIFY_IP4_ADDRESS

first ip4 address in the list

VERIFY_IP4_PORT

VERIFY_IP6_ADDRESS

first ip6 address in the list

VERIFY_IP6_PORT

Configuring dnS2exy

server:

```
verifier-count:  <# concurrently running verifiers>
verifier-timeout: <# seconds>
verify-ip-address: <ip4 or ip6>[@port]
verify-port:  <number>
verify-feed-zone:  <yes or no>
```

zone:

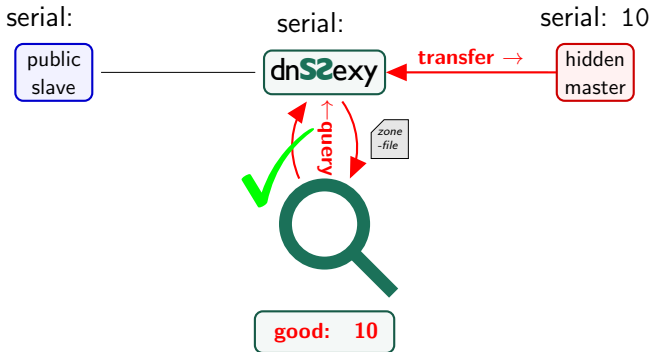
```
verifier:  <program with arguments>
verifier-timeout:  <# seconds | inherit>
verifier-feed-zone:  <yes, no or inherit>
```

VERIFY_ZONE	name of the changed zone	
VERIFY_IP_ADDRESSES	list of <address@port> values	
VERIFY_IP_ADDRESS	first ip address in the list	VERIFY_PORT
VERIFY_IP4_ADDRESS	first ip4 address in the list	VERIFY_IP4_PORT
VERIFY_IP6_ADDRESS	first ip6 address in the list	VERIFY_IP6_PORT
VERIFY_ZONE_ON_STDIN	yes or not set	

Operational details



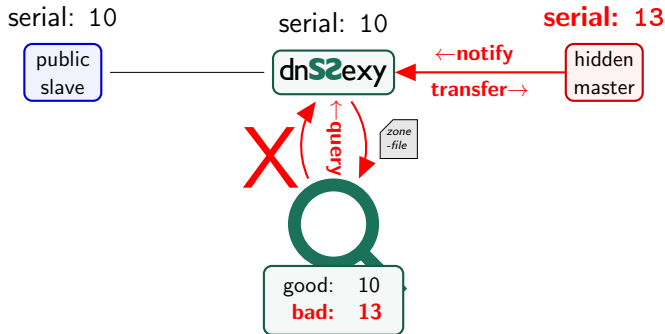
Operational details



Operational details



Operational details



Operational details

- ▶ Bad serials will not be transferred again ...



Operational details

- ▶ Bad serials will not be transferred again ...



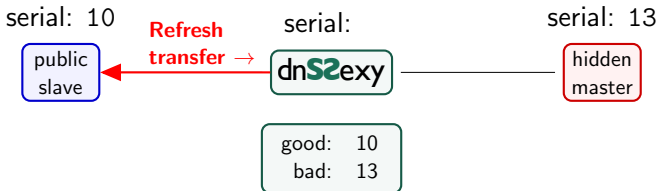
Operational details

- ▶ Bad serials will not be transferred again ...



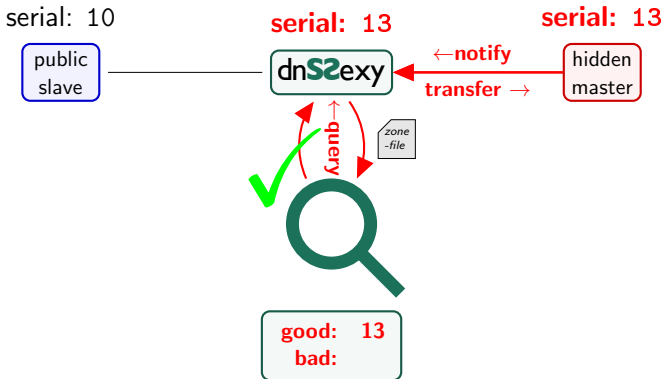
Operational details

- ▶ Bad serials will not be transferred again ...
- ▶ dnS2exy provides transfer for expired zones (already in NSD).
 - ▶ add option **provide-xfr-when-expired:** <yes or no>
 - ▶ add option **serve-when-expired:** <yes or no>



Operational details

- ▶ Bad serials will not be transferred again ...
 - ▶ RRSIGs may become valid in the future
- ▶ ... unless notified by master



Status

- ▶ dnS2exy has just been code-reviewed within NLnet Labs
- ▶ Beta release candidate will follow shortly ...
- ▶ ... with the beta release one week later if all is well
- ▶ Subscribe to the mailinglist:
<https://www.NLnetLabs.nl/mailman/listinfo/dnssexy>
- ▶ Spot on the web will (probably) become:
<https://www.NLnetLabs.nl/project/dnssexy>
- ▶ Or from the subversion repository:
`svn co https://www.NLnetLabs.nl/svn/nsd/branches/NSD_3_2_SEXY`

Trying out dnS2exy

- ▶ branches/NSD_3_2_SEXY/tpkg/dnssexy-setup
- ▶ Used by the unit tests
- ▶ Configure either in dnssexy-setup directory, or in another subdirectory of tpkg or have dnS2exy installed
- ▶ Demo?

Future improvements

► dnS2exy on the side

Future improvements

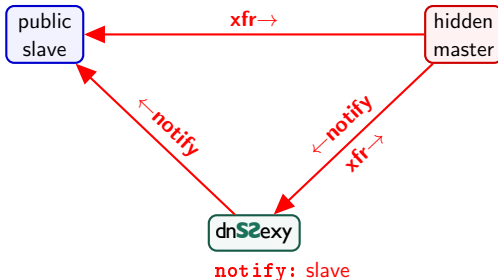
► dnS2exy on the side

- **NSD** has no support for IXFR out
- Verifiers could trash the proxy
- Notices and transfers are configured independently with **NSD**

in:	allow-notify:	request-xfr:
out:	notify:	provide-xfr:

```
allow-notify: dnS2exy
request-xfr: master
```

```
notify: dnS2exy
provide-xfr: dnS2exy
provide-xfr: slave
```

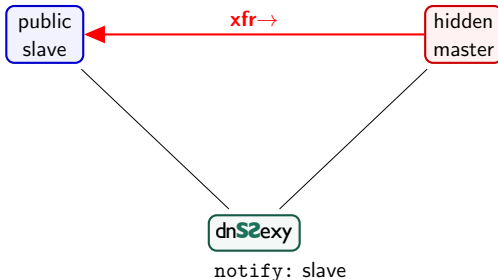


Future improvements

- ▶ dnS2exy on the side
 - ▶ Slave could transfer bad zones on Refresh and Retry!

allow-notify: dnS2exy
request-xfr: master

notify: dnS2exy
provide-xfr: dnS2exy
provide-xfr: slave

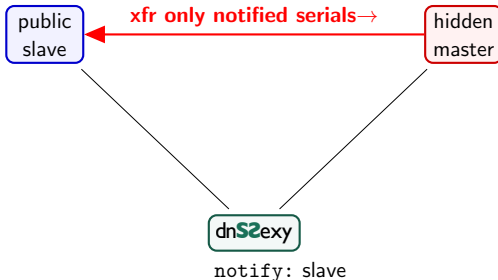


Future improvements

- ▶ dnS2exy on the side
 - ▶ Slave could transfer bad zones on Refresh and Retry!
 - ▶ xfr-only-notified-serials: <yes or no>

```
allow-notify: dnS2exy
request-xfr: master
xfr-only-notified-serials: yes
```

```
notify: dnS2exy
provide-xfr: dnS2exy
provide-xfr: slave
```

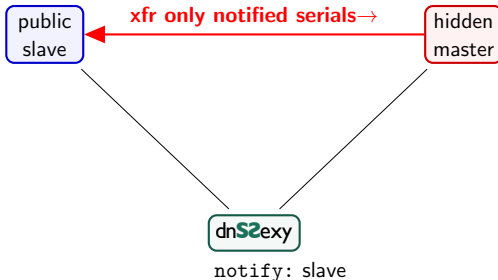


Future improvements

- ▶ dnS2exy on the side
 - ▶ Slave could transfer bad zones on Refresh and Retry!
 - ▶ xfr-only-notified-serials: <yes or no>
 - ▶ Bootstrap problem!

```
allow-notify: dnS2exy
request-xfr: master
xfr-only-notified-serials: yes
```

```
notify: dnS2exy
provide-xfr: dnS2exy
provide-xfr: slave
```

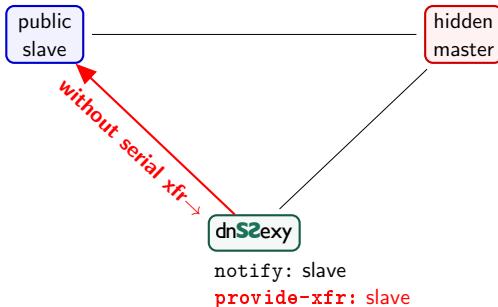


Future improvements

- ▶ dnS2exy on the side
 - ▶ Slave could transfer bad zones on Refresh and Retry!
 - ▶ xfr-only-notified-serials: <yes or no>
 - ▶ Bootstrap problem!
 - ▶ without-serial-request-xfr: <ip> <key | NOKEY>

```
allow-notify: dnS2exy
request-xfr: master
xfr-only-notified-serials: yes
without-serial-request-xfr: dnS2exy
```

```
notify: dnS2exy
provide-xfr: dnS2exy
provide-xfr: slave
```



```
notify: slave
provide-xfr: slave
```


Future improvements

- ▶ dnS2exy on the side
 - ▶ `xfr-only-notified-serials:` <yes or no>
 - ▶ `without-serial-request-xfr:` <ip> <key | NOKEY>
- ▶ dnS2exy is authoritative for valid zones
 - ▶ `provide-xfr-when-expired:` <yes or no>
 - ▶ `serve-when-expired:` <yes or no>
- ▶ Verifiers could query the master directly
 - ▶ Environment variable with the master from which the xfr came
`VERIFY_XFR_RECEIVED_FROM=<ip@port>`
- ▶ Feeding the verifier with changed part only
 - ▶ `verifier-feed-zone:` <yes, **incremental** or no>
- ▶ Verify zones loaded from zone files too

Future improvements

- ▶ dnS2exy on the side
 - ▶ xfr-only-notified-serials: <yes or no>
 - ▶ without-serial-request-xfr: <ip> <key | NOKEY>
- ▶ dnS2exy is authoritative for valid zones
 - ▶ provide-xfr-when-expired: <yes or no>
 - ▶ serve-when-expired: <yes or no>
- ▶ Verifiers could query the master directly
 - ▶ Environment variable with the master from which the xfr came
VERIFY_XFR_RECEIVED_FROM="<ip@port>"
- ▶ Feeding the verifier with changed part only
 - ▶ verifier-feed-zone: <yes, **incremental** or no>
- ▶ Verify zones loaded from zone files too

Future improvements

- ▶ dnS2exy on the side
 - ▶ `xfr-only-notified-serials:` <yes or no>
 - ▶ `without-serial-request-xfr:` <ip> <key | NOKEY>
- ▶ dnS2exy is authoritative for valid zones
 - ▶ `provide-xfr-when-expired:` <yes or no>
 - ▶ `serve-when-expired:` <yes or no>
- ▶ Verifiers could query the master directly
 - ▶ Environment variable with the master from which the xfr came
`VERIFY_XFR_RECEIVED_FROM="<ip@port>"`
- ▶ Feeding the verifier with changed part only
 - ▶ `verifier-feed-zone:` <yes, incremental or no>
- ▶ Verify zones loaded from zone files too

Future improvements

- ▶ dnS2exy on the side
 - ▶ xfr-only-notified-serials: <yes or no>
 - ▶ without-serial-request-xfr: <ip> <key | NOKEY>
- ▶ dnS2exy is authoritative for valid zones
 - ▶ provide-xfr-when-expired: <yes or no>
 - ▶ serve-when-expired: <yes or no>
- ▶ Verifiers could query the master directly
 - ▶ Environment variable with the master from which the xfr came
VERIFY_XFR_RECEIVED_FROM="<ip@port>"
- ▶ Feeding the verifier with changed part only
 - ▶ verifier-feed-zone: <yes, incremental or no>
- ▶ Verify zones loaded from zone files too

Future improvements

- ▶ dnS2exy on the side
 - ▶ `xfr-only-notified-serials:` <yes or no>
 - ▶ `without-serial-request-xfr:` <ip> <key | NOKEY>
- ▶ dnS2exy is authoritative for valid zones
 - ▶ `provide-xfr-when-expired:` <yes or no>
 - ▶ `serve-when-expired:` <yes or no>
- ▶ Verifiers could query the master directly
 - ▶ Environment variable with the master from which the xfr came
`VERIFY_XFR_RECEIVED_FROM="<ip@port>"`
- ▶ Feeding the verifier with changed part only
 - ▶ `verifier-feed-zone:` <yes, **incremental** or no>
- ▶ Verify zones loaded from zone files too
- ▶ Suggestions?

Summary

subversion	https://www.NLnetLabs.nl/svn/nsd/branches/NSD_3_2_SEXY
mailing-list	https://www.NLnetLabs.nl/mailman/listinfo/dnssexy
web	https://www.NLnetLabs.nl/project/dnssexy
me	mailto:Willem@NLnetLabs.nl

What is dnS2exy

- ▶ A building block for fortifying DNSSEC
- ▶ A hook opening up new possibilities in provisioning strategies

Question

Who is more authoritative? The master or the validator?

Summary

subversion https://www.NLnetLabs.nl/svn/nsd/branches/NSD_3_2_SEXY
mailing-list <https://www.NLnetLabs.nl/mailman/listinfo/dnssexy>
web <https://www.NLnetLabs.nl/project/dnssexy>
me <mailto:Willem@NLnetLabs.nl>

What is dnS2exy

- ▶ A building block for fortifying DNSSEC
- ▶ A hook opening up new possibilities in provisioning strategies

Question

Who is more authoritative? The master or the validator?

